

Les 1 software analyse

iljo De Poorter

September 28, 2023

1 Intro

Waar gaat het vak over? Invullen in de avond

Kwaliteitsvolle producten leveren aan de klant. Wat wil de klant hebben? – requirements. Wij moeten capteren wat functioneel maar ook niet functioneel nodig.

Wanneer is iets succesvol. – requirement, budget(erbinnen) en op tijd geleverd. Als we alle drie van de criteria voldoen was het succesvol.

Maar in het echt is dat wel niet zo..... Veel mislukt. Dit kan voorkomen worden door meer op voorhand te plannen en te weten wat je precies wilt. Om zo een goede prijs te kunnen geven.

Dus communicatie is belangrijk.

Zowel met de klant, de collega's en de Business zelf.

1.1 leerinhoud

Welk sw-ontwikkelingsproces zullen wij hanteren?

Wat zijn de requirements voor een project? Dit krijgen we van de leerkracht.

Dit moeten wij dan omzetten in diagrammen voor de SW developer.

Systeem sequentie diagram.

Niet functionele requirements.

1.2 SW-ontwikkeling vb1

Nadat de klant zijn eisen geeft moeten wij verder? wat doen wij?

Wat te doen nadat we onze requirements hebben van de klant? Bekijken of het realistisch of wat de klant wilt? Specifieke eisen specificeren. Prioriteiten stellen.

Goed luisteren naar wat de klant wilt, en dan verder gaan met de analyse.

Elk ontwikkelingsproces bestaat de volgende onderdelen? Requirements, verzamelen door te luisteren naar de opdrachtgever.

1.3 SW-proces

Requirements analyseren.

Een ontwerp maken

Implementeren, het uitvoeren van ons ontwerp testen.

1.4 testcase 1/watervalmethode

Nieuwe website maken voor een e-gaming event in juli. We starten op 1 oktober.

We werken volgens de SW-process methode

Waarom de waterval methode slecht is

- Doordat alles in sequentie gebeurt verlies je veel tijd.
- Je kan niet terug naar een vorige fase.
- terwijl je dan bezig bent met het coderen heb je geen communicatie met de klant, aangezien alles NA elkaar gebeurt.

Doordat we alles NA elkaar doen verliezen we veel tijd en doen we veel slecht. Dit moeten we dus oplossen door AGILE toe te passen.

1.5 agile

Een snellere, betere manier om SW te doen. Er is een statistisch hogere kans dat je project zal werken. Werkt in 2 methodes Iteratief en incrementeel. Door iteratief te werken werken we in "lussen". We herhalen dezelfde stappen oconstant, maar tussen alle stappen door praten we met de klant en vragen we feedback. En door het incrementel te doen gaan we ook snel vooruit. We maken dus iets en bouwen er op verder, incrementeel.

Door niet incrementeel te werken(Niet met elkaar te praten en te verbeteren) loopt het vaak mis.

1.5.1 meer uitleg over agile

Meer flexibiliteit en wendbaarheid(Vs. Waterval) Doordat we constant feedback vragen in korte stukken kunnen we sneller aanpassingen maken. Kleine delen herwerken is sneller dan grote blokken herwerken.

Iteratief, je werkt in stappen, na elke stap lever je "werkende software" op, waarmee de gebruiker kan werken en krijg je feedback.

Prioriteiten is het 'ruwe' plan aanhouden, maar nog liever houden we gewoon de klant tevreden.

1.5.2 Iteratief-incrementele ontwikkeling

- Think big, develop smal, werk in kleine iteraties

- Een iteratie bevat steeds dezelfde activiteiten met feedback en extra's om vooruit te gaan.
- De tijdsbesteding aan iedere activiteit is niet gelijk.
- Iteraties duren van 2-6 weken, dit is NIET strict.

Er is minder failure bij agile(In vergelijking met Waterval) is omdat we constant feedback vragen en zo kunnen we snel aanpassingen maken.

1.5.3 UML

Unified Modeling Language Modeleertaal die gebruikt wordt voor objectgeoriënteerde analyse en ontwerp. Kan gebruikt worden voor eigenlijk elk soort ontwerp(Agile, waterval) Dit is een notatie wijze. Voordelen

- Communicatie
- Visualisatie
- Transformatie
- -analyse, ontwerp
- -ontwerp, programmeren
- -programmeren, testen

2 Domeinen?

Wie heeft heeft interactie met elkaar? Hoe zit alles in elkaar? SQL, wat zijn de verbanden met elkaar?